PATENT

Patent Application of

Robert M. Best

for

TITLE: Video Game System with Diverse Controllers

Suggested class/subclass: 463-42

RELATED APPLICATION

This application is a continuation-in-part of copending U.S. patent application Ser. Num. 09/853,487 filed May 10, 2001 and titled "Video Game System with LCD Controllers" which application in its entirety is incorporated herein by reference.

BACKGROUND – FIELD OF INVENTION

This invention relates generally to electronic video game systems and more particularly to electronic video game systems that have handheld control units with liquid-crystal display (LCD) screens.

BACKGROUND – DISCUSSION OF PRIOR ART

Video game console systems, handheld control units, and handheld electronic games having liquid crystal display (LCD) screens are well known and are described in US patent 5,393,073. It is also known to distribute video games on plastic discs on which encrypted information has been written for verifying authenticity. It is also known to use touch-sensitive screens and touchpads, in addition to or in place of a mouse, for

entering information into handheld computers. It is also known to use analog joysticks to manipulate movement of player controlled characters in simulated 3-dimensional space (see US patent 6,139,433) on a TV-screen. It is also known for a human game player to control groups of multiple player-controlled characters (see the Nintendo game Pikmin). It is also known for a player to request that his player-controlled character perform an action or task directed toward an object (see The Sims game). It is also known to display secret information on handheld control units to hide the information from other players.

In a video game in which two or more human players control their respective player-controlled characters on a TV-screen using handheld controllers with LCD screens (see my US patent 5,358,259), a problem arises as to how each human player can signal to the game console (the game system's main computer) what the player wants his/her character to do, other than using push buttons and joysticks to control simple actions such as running, jumping, hitting, shooting, etc. In a multi-player game, some of the selected and rejected actions for a player's character should not be seen on the TV screen by other players. A human player can indicate his/her wants by making a selection on a handheld menu of words, but this is not very natural.

Patent application GB 2,353,928A discloses a game system having a console connected to multiple handheld game machines with LCD's that display maps including squares to indicate player-controlled characters, circles to indicate monsters, and diamonds to indicate items. Although this patent maintains that these maps are pictures, the patent does not provide any examples of pictures of animated characters with hands, arms, legs, faces, and clothing for display on handheld control units.

Therefore, a need has arisen for handheld controllers that display more natural visual information such as pictures, especially pictures of characters, that enable players to control

their TV-screen characters more naturally than with prior-art controllers.

SUMMARY

An embodiment of this invention is a video game system that includes a console unit and handheld control units. The console unit generates animated pictures for display on a television (TV) screen. Handheld control units are of two kinds: those that include an LCD screen that displays pictures, maps, words, numbers, etc. and control units that do not include LCD screens. Players may use both kinds of control units at the same time. The pictures may be still pictures and/or animated pictures. During parts of the game, each control unit may directly control animated player-controlled characters that are displayed on the TV screen, and control units that have LCD screens can display pictures of scenes and animated characters that are different from the scenes and characters displayed on the TV screen. Each LCD control unit may operate for awhile as a personal game unit while remaining in coordination with the console game unit that may be generating pictures of the same scene or a different scene for display on the TV screen. Pictures displayed on a control unit LCD screen may appear concurrently or later on a TV screen.

Simulated objects and characters are displayed on the LCD screen of a control unit in a natural pictorial setting, to the degree that it is possible to make pictures look natural on an LCD screen, and can be selected, moved, constructed, changed, or deleted by a player without revealing to other players these objects of interest or their disposition.

In the preferred embodiment, each player uses two control units: one handheld control unit that has one or more joysticks and/or touchpads for controlling player-controlled characters in a simulated three-dimensional world, and a second control unit with an LCD screen so that players can view pictorial information that is hidden from other players and select and control objects,

characters, actions, and tasks on the LCD screen. The video game system in general will provide a unified game experience in which a combination of controllers do more than just control a console game, but also do more than just a stand-alone handheld game.

Each game operates in a simulated world populated with animated characters and static objects which are displayed on the screen of the TV set, and may also be displayed on controllers with LCD screens. While one part of the simulated world is displayed on the TV screen, different parts of the simulated world may appear on LCD screens while each player uses a handheld control unit to control one or more player-controlled characters on the TV screen or LCD screen or both. Some of the pictures displayed on LCD screens and TV screens may represent the same part of the simulated world at different times, or different parts at the same time, or the same part at the same time.

In a war game for example, while a first player is controlling a soldier fighting a skirmish in one part of the simulated world that appears on the first player's LCD screen, a second player may be controlling a different character building a fortification in a different part of the simulated world and this building scene appears on the second player's LCD screen, while a third part of the simulated world appears on the TV screen in this example. Later, the TV screen may display the fortification that was secretly built by the second player's character.

Each player may control more than one player-controlled character and may assign tasks to each character. For example the task of building a fortification may be assigned to a team of player-controlled characters, each of which can be individually controlled by a player operating one or two control units, within the overall preprogrammed task of building the fortification. A player may assign a task to a robot character who then performs his task on the TV screen or LCD screen without the player having

to guide each movement. A player may intervene and control a task to whatever degree is necessary. Movements of some player-controlled characters may be controlled by two or more players simultaneously.

ADVANTAGES

By displaying pictures on an LCD screen for each player, alternative dispositions of objects and characters in the game are presented to players in a natural setting, unlike menus of words or symbols representing characters. This reduces clutter on the TV screen which might otherwise reveal to other players unfinished tasks or hidden alternatives or selections. Natural pictures on an LCD screen will provide quicker and more accurate recognition and selection of locations, directions, orientation, and actions of game characters before they appear on the TV screen.

OBJECTIVES

An object of this invention is to make strategy and role-playing video games more fun for players by providing alternative choices for each player in personalized natural pictures on control units so that control information does not clutter the main TV pictures and that players' confidential alternatives or selections are not revealed to other players on a TV screen.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 shows an exemplary game playing session in which two human game players hold game control units having LCD screens on which are displayed miniature copies or likenesses of large pictures displayed on the screen of a television set.

Fig. 2 shows an exemplary game playing session in which two human game players hold game control units having LCD screens on which are displayed respectively a miniature copy or likeness of the large TV picture and a miniature preview picture of a later scene.

Fig. 3 is an external isometric view of an exemplary handheld control unit including an LCD screen and touch-sensitive pad.

Fig. 4 is a block diagram of the Fig. 3 control unit.

Fig. 5 is an isometric view of a handheld control unit illustrating manual selection of numbers by using a cross-switch and a push-button.

Fig. 6 is an isometric view of a handheld control unit with a touch-sensitive LCD screen illustrating manual selection of numbers.

Fig. 7 is an isometric view of a handheld control unit with a touch-sensitive LCD screen illustrating manually controlled movement of a selected picture object.

Fig. 8 is an isometric view of an exemplary video game system using two of the Fig. 3 control units.

Fig. 9 is an isometric view of a prior-art video game system from Fig. 9 in US Patent 5,358,259.

Fig. 10 is an isometric view of a handheld control unit with a touch-sensitive LCD screen illustrating manual selection of character emotions.

Fig. 11 is a touch-sensitive LCD screen with cartesian coordinates superimposed to illustrate selection and movement of simulated objects in two dimensions on an LCD picture.

Fig. 12 is a map on an LCD screen to illustrate manual selection of a line segment defined by a pair of 2-dimensional locations on the map.

Fig. 13 is a map on an LCD screen to illustrate a line of soldiers in a war game.

Fig. 14 is a map on an LCD screen to illustrate creation of a simulated barrier on a bridge in a war game.

Fig. 15 is a touch-sensitive LCD screen illustrating manual selection of an action to be performed by a game character from four alternative actions.

Fig. 15a is a series of LCD pictures for manual selection of an action to be performed by a game character from more than two alternative actions.

Fig. 16 is a block diagram of an exemplary video game system having two handheld control units.

Fig. 17 is a block diagram of the Fig. 16 video game system with details of an exemplary security processor chip.

Fig. 18 is a block diagram of a disk manufacturer's process of encrypting data and writing it onto an optical disk.

Fig. 19 is a record format indicating various data fields in a location data record.

Fig. 20 is a memory map of various programs stored in a handheld control unit.

Fig. 21 is a flow chart of program processes in a handheld control unit.

Fig. 22 is a TV screen displaying a picture of a video game scene to illustrate levels of detail that may occur in a picture.

Fig. 23a, 23b, and 23c are an LCD screen displaying a likeness of the picture in Fig. 22 but greatly reduced in size.

Fig. 24 is a simplified block diagram of the system showing how data flows between the console and a handheld control unit.

Fig. 25 is a flow chart of program processes in a handheld control unit.

Fig. 26 shows an exemplary game playing session in which two human game players each hold a game control unit and each player receives visual information from a TV screen and from a second game control unit having an LCD screen.

Fig. 27 is a map view of two cameras focused on different objects and generating pictures for different display devices.

Fig. 28 is an LCD display screen displaying a scene from a simulated game world that is positioned between rows of graphic control information.

Fig. 28a is an LCD screen displaying a picture menu of characters for point of view selection.

Fig. 29 is a cross-sectional map view of a cave in which a robot camera is focused on a hidden object that is not observable from the point of view of a player-controlled character.

Fig. 30 is a side view of a flying robot for games that remotely control a simulated robot with grippers.

Fig. 31 is a perspective view of a land crawling robot for games that remotely control a simulated robot with grippers.

Fig. 32 is an LCD screen displaying information for assigning control members to control various robot movements.

Fig. 33 is an LCD screen displaying information for activating or deactivating a character or characters.

Fig. 34 is an LCD screen displaying a picture menu for selecting a task for a character to perform.

Fig. 34a is an LCD screen displaying information for assigning a task to a character.

Fig. 35 is a flowchart of program processes in a games with characters that are player-controlled and task-controlled.

Fig. 36 shows an exemplary game playing session in which two human game players each hold a game control unit and each player receives visual information from a TV screen and from a second game control unit having an LCD screen.

Fig. 37 is an isometric view of a hinged assembly for securing two LCD control units.

Fig. 38 is an isometric view of a non-hinged support assembly for securing two LCD control units.

Fig. 39 is a map view showing pictures of objects generated from different points of view.

Fig. 40 is a memory map of programs and data.

Fig. 41 is a game system with two control units illustrating trading of objects.

Fig. 42 is a three dimensional (x,y,z) graph illustrating cartesian coordinates of cameras and an object being viewed.

Fig. 43 shows an exemplary game playing session in which a human game player operates three control units: two units with LCD screens and one unit with no LCD screen.

Fig. 44 shows an exemplary game playing session in which two human game players both control the same player-controlled character, a simulated robot.

Fig. 45 is an orthographic projection of an adapter used to add control functions to a portable game unit.

Fig. 46 is a block diagram of the Fig. 45 adapter used with the portable game unit.

DETAILED DESCRIPTION OF EXEMPLARY EMBODIMENTS

Fig. 8 shows an exemplary embodiment of a video game system 118 on which the video games of the present invention may be played. Video game system console 42 generates a video signal on cable 41 connected to TV set 11, for display on TV screen 56 or on a video monitor (not shown) or similar common display seen by other players. Console 42 is also connected to one or more handheld control units 28 and 29 or other user input devices by cables 45 or a wireless equivalent (not shown in Fig. 8) such as infrared, ultrasonic, RF waves, or other data communicating forms of energy. Console 42 is detailed in Fig. 16 which shows an optical disk reader 83 for reading optical disks 43 in which tracks 82 of digital information, including game programs and data, are pressed and molded by a disk manufacturer.

The improved control units 28 and 29 shown in Fig. 8 and Fig. 3 (control unit 29 is the same design as unit 28) include features not included in control units 44 and 47 shown in other drawings. This is done for clarity in the drawings and does not imply that any one control unit design is more or less suitable for the present invention, except where additional hardware features of control units 28 and 29, such as touch pad 24 and touch screen 23, are required for use in video games that make use of those hardware features.

Fig. 1 illustrates an exemplary game playing session in which two human game players 10 and 12 hold game control units 44 and 47 having LCD screens on which are displayed pictures, verbal expressions, and/or other visual images. Whenever a human player 10 presses push-button (button-switch) 14, his handheld control unit 44 generates on his LCD screen a miniature copy 33 of the large picture displayed on TV screen 56, generated either from data already stored in control unit 44, or from data transmitted from console 42 (Fig. 16) in response to a signal initiated by manually pressing button 14. Miniature picture 33

may be a freeze-frame, or animated in sync with the TV picture at various display rates, or animated in slow or accelerated motion.

After miniature picture 33 is displayed on the LCD screen of control unit 44, one or more areas 25 of the LCD screen may blink or change color or brightness or otherwise highlight or indicate areas of possible interest to player 10. Player 10 may select a simulated object or area in picture 33 for further study by using cross-switch 15 to position a cursor, highlight, or other visual indicator to an LCD screen location corresponding to the indicated area 25. Player 10 then selects the object or indicated location by pressing selection push-button 57, which may cause the indicated area 25 to be enlarged on the LCD screen as picture 34 so that an object 31 that was previously invisible or too small to see on the LCD screen is made visible. Player 10 may then repeat the process by selecting object 31 which may be a written clue (with words that appear on control unit 44) or a weapon to keep for future action, or other selectable objects. When objects are highlighted or enlarged on unit 44, they typically are not highlighted or enlarged on TV screen 56 so that other human players such as player 12 will not see which objects have been selected on unit 44.

Alternatively, player 10, who does not normally control the dinosaur, may select the dinosaur's foot 58 that is blinking or otherwise indicated on the LCD screen of control unit 44. When player 10 positions a cursor or other location indicator on foot 58 and presses selection button 57, the action sequence of digitally generated pictures being displayed on TV screen 56 may, for example, cut to an alternative action sequence showing the dinosaur stumbling and falling accompanied by sounds of the dinosaur hitting the ground and screaming in pain and anger, thereby allowing character 17 to escape from the dinosaur.

During the time that player 10 is pressing cross-switch 15 and buttons 14 and 57, the action sequence showing the dinosaur

chasing character 17 will continue and may reach a different branch point in the branching structure of action sequences that makes player 10's selections moot. For example, player 12 may be making alternative choices that display different objects of interest on her control unit 47 and she may select different branches in the branching structure of action sequences that display alternative actions of character 17 or the dinosaur, or alternative scenes and characters.

Role-playing video games that make use of this invention will typically promote both cooperation and competition between game players. The exemplary game may promote cooperation between players 10 and 12 in trying to stop the dinosaur from attacking character 17, but the game may also create competition between players 10 and 12, both of whom may want to be first to rescue character 17.

In many embodiments, miniature picture 33 is a freeze frame so that human player 10 may select an object 25 on the LCD screen before the object moves off screen.

Fig. 2 illustrates an exemplary game playing session in which human player 10 has selected the miniature picture option described above with reference to Fig. 1 and has positioned cursor 49 onto the hand 36 of his player controlled character. The cursor appears only on miniature picture 33 and not on TV screen 56. Player 10 has selected on his control unit 44 a hand-control mode in which he can control 3-dimensional movement of the hand of his player-controlled character. In the preferred control unit design shown in Fig. 3, handheld control unit 28 includes at least one analog joystick 20 or 21 by which player 10 in Fig. 2 may control 3-dimensional movement of his player-controlled character's right hand 36 or other selected body part. Details of a 2-shaft analog joystick to control motions of a player controlled character in 3-dimensions are disclosed in US patent 6,186,896.

In the exemplary game illustrated in Fig. 2, player 10 has used cross-switch 15 to position his player character's right hand 36 to grasp steel pipe 35 for use as a prybar to open the door of a wrecked car shown in miniature picture 33 on the LCD screen of control unit 44. When player 10 selects this option, his control unit 44 sends a data record (Fig. 19) to console 42 (Fig. 8) requesting a hand-grasping action sequence, and console 42 responds by generating a video frame sequence combining rendered polygons of moving hand 36 superimposed on the wrecked car background. Console 42 also generates a video signal for the generated frame sequence for display on TV screen 56 so that the other player 12 may see the hand-grasping action.

Simultaneously, control unit 44 generates an equivalent sequence of miniature animated pictures of moving hand 36 superimposed on the same wrecked car background on the LCD screen of control unit 44. After the sequence of miniature animated pictures 33 and the frame sequence of video pictures shown on TV screen 56 begin, both sequences continue and remain substantially in sync, although perhaps at a different display rate, until player 10 selects other images for viewing on his control unit 44, or another player 12 alters the moving picture sequence on TV screen 56. The moving pictures on TV screen 56 of hand 36 grasping pipe 35 are visible to other human player 12 with no indication on TV screen 56 that any cursor control was used to cause the hand-grasping action sequence.

Human player 12 has selected (as will be explained below with reference to Fig. 15) an action from a picture menu (Fig. 15 or 15a) of alternative actions displayed on her control unit 47. This selected action enables player 12 to position her cursor 59 (Fig. 2) on the right hand 37 of her player-controlled character to add her character's simulated pulling force to pipe 35. When player 12 selects an action from a picture menu, her control unit 47 displays a miniature preview picture 34 on the LCD of her

control unit 47 showing what will happen if she implements her selected action.

To accomplish this, her control unit 47 generates and displays an action sequence showing two hands 59 and 36 successfully pulling on pipe 35. This preview sequence can be generated in simplified, low-resolution, fast-motion form, to give player 12 a quick preview of the selected (but not yet implemented) action sequence that will appear on TV screen 56 if she implements it.

In the exemplary Fig. 2 game, if player 12 implements the selected action by pressing on an appropriate push-button, her control unit 47 sends a selection data record (Fig. 19) to console 42 (Fig. 8) which generates the frame sequence being displayed on TV screen 56 and will, for example, generate a modified frame sequence showing her player-controlled character's right hand 37 grasping pipe 35 beside the other character's right hand 36 followed by a picture sequence showing both player-controlled characters prying open the wrecked car door and rescuing a non-player character (not shown) in the wrecked car.

Likewise in Fig. 1, player 10 may rerun prior scene 34 on LCD 22 so that he may make use of clue 31 or pickup tools he neglected earlier. Button-switches 14 may provide rewind, normal speed, and fast forward control of pictures displayed on LCD 22 for manual selection of objects and clues from prior scenes.

Fig. 3 shows an improved handheld control unit 28 which overcomes some of the difficulties a player might have selecting actions and objects on an LCD screen using only cross-switch 15 and push-buttons 14 and 57 on the handheld control units 44 and 47 illustrated in Fig. 1 and Fig. 2. The exemplary Fig. 3 control unit includes cross-switch 15, two analog joysticks 20 and 21, push-buttons 57, 14 and other buttons, speaker 27,

external memory cartridge 16, touch-sensitive pad 24, and LCD 22 covered with transparent touchscreen 23 (shown in Fig. 4).

Touchpad 24 and touchscreen 23 are sensitive to finger pressure and can measure the approximate location of a finger on X-Y coordinates as described below with reference to Fig. 11. Transparent touchscreen technology is described in US patent 6,163,313. In Fig. 3 herein, both touchpad 24 and touchscreen 23 are specified for control unit 28 so that a player can use fingers of both hands to maneuver virtual objects in 3-dimensional space on LCD screen 22. A player can select an object on touchscreen 23 with one finger, and while holding his finger steadily on the object, use another finger on touchpad 24 to rotate the object into the desired position. Touchpad 24 and touchscreen 23 can also act as push-buttons by accepting a finger tap, for example, of a few hundred milliseconds as a selection indicator.

Fig. 4 is a block diagram of the Fig. 3 control unit 28 which connects to console 42 through connector 40 and cable 45 or wireless equivalent. Control unit 28 which is only schematically represented in Fig. 4 includes touchscreen 23, touchpad 24, and controller processor 51 for determining finger locations on touchscreen 23 and touchpad 24. Processor 51 outputs X and Y coordinates to processor 50 which generates all pictures and text that appear on LCD 22 via LCD driver 119, and generates data records (Fig. 19) that processor 50 sends to console 42. Processor 50 also interprets all data records received from console 42 including records containing data from which processor generates pictures for display on LCD 22. Memory security processor 52 controls all data passing between processor 50 and external memory cartridge 16 to verify authenticity of cartridge 16. Memory cartridge security processors are disclosed in US patent 6,190,257. Memory cartridge 16 is typically used when control unit 28 is used as a stand-alone handheld game system.

When electric power to control unit 28 is turned on, boot ROM 76 provides an initial program of instructions, including some programs listed in Fig. 20. Additional programs are loaded into RAM 53 and are supplied by console 42 which reads these control unit programs from disk 43. See further discussion of these programs below with reference to Figures 19, 20, and 21.

Control unit 28 may include various other features such as an operating system in ROM 76, a ROM and battery-maintained RAM in external memory cartridge 16, a data bus, an address bus, input/output processor, image processing unit, communication control unit, power source, circuit board, and other customary components.

Fig. 5 illustrates a slow method of entering numbers, without using a keyboard, by pressing cross-switch 15 repeatedly to move highlight cursor 48 horizontally and vertically on LCD screen 22 until a desired digit is highlighted. Pressing button 57 enters the selected digit. After all digits have been entered, button 57 is pressed again to enter the multi-digit number. This method is often too slow for games that require entering numbers, such as map coordinates for war games. Using analog joystick 20 is typically faster but less accurate, because pressing the joystick a little too far causes the highlight cursor to overshoot the desired digit.

Fig. 6 illustrates a faster method of entering digits using touchscreen 23 overlaying LCD 22. After selecting a series of digits by touching the digits, button 57 is pressed only once to enter the multi-digit number. For games that are downloaded from the Internet after payment by credit card, the touchscreen method illustrated in Fig. 6, for entering credit card numbers, is the preferred method, because entry of such numbers can be easily kept hidden from other people when entered on a handheld control unit. Connector 40 for communications between control unit 47

and game console 42 may be connected to wires in cable 45, or an RF transceiver, or a transceiver using infrared photodiodes 38.

Fig. 7 illustrates use of touchscreen 23 to replace the cursor control described above with reference to Fig. 2. Instead of using cross-switch 15 in Fig. 2 to position cursor 49 on hand 36 or cursor 59 on hand 37, the preferred method in Fig. 7 is for human player 12 to touch her finger to touchscreen 23 overlying the LCD image of hand 37 and slide her finger across touchscreen 23 to a new location over pipe 35 to cause corresponding movement of hand 37 grasping pipe 35. Touchscreen 23 signals the finger location to controller 51 (Fig. 4), which converts the location to physical X,Y coordinates, which processor 50 uses to calculate a new LCD location for displaying hand 37. Thus simulated hand 37 will follow the player's moving finger on the touchscreen without any need for a cursor. The image of hand 37 substitutes for a cursor. When the location of hand 37 is within preprogrammed coordinates for pipe 35, processor 50 (Fig. 4) recomputes the pixels representing hand 37 in successive frames, so that the hand appears to grasp and move pipe 35 displayed on the LCD. See further discussion below with reference to Fig. 11.

Processor 50 also sends a series of data records to console 42 selecting a branch in the branching structure of alternative sequences of hand movements, showing hand 37 moving to the location of pipe 35, rotating to a new angle facing pipe 35, and grasping pipe 35, the image of which is separately generated with the corresponding size and orientation. Microprocessor 86 (Fig. 16) or graphics coprocessor (not shown) in console 42 then generates the corresponding sequence of rendered polygons for hand 37 and pipe 35 for including in the video frame sequence. With this Fig. 7 method, players can use their handheld controllers to indicate movement of objects to new locations in 3-dimensions and indicate actions to be performed which are then typically generated as composite video by generator 117 (Fig. 16) and appear on TV screen 56 for both players 10 and 12 to see.

Fig 8 shows an exemplary video game system 118 in general which includes two of the improved control units 28 and 29 as described above with reference to Fig. 3.

Prior-art hardware shown in Fig. 9 (from my US patent 5,358,259) is included herein for comparison with Fig. 8. LCD screens 22 are illustrated in Fig. 8 showing pictures, in contrast with Fig. 9 LCD screens 13 which show menus of verbal expressions. For clarity, other differences in hardware, software, and methods are not all shown in Figs. 8 and 9.

Fig. 10 shows a control unit 47 with touchscreen 23 and a picture menu of emotional faces. By touching one face 32, human player 12 can select the desired emotion or mood of a player-controlled character.

Fig. 11 illustrates manual use of touchscreen 23 with X,Y coordinates for indicating a two-dimensional location on the underlying LCD screen 22 (Fig. 4). Fig. 11 shows hand 37 shaped as a fist and located at coordinates (X_1 Y_1). When human player 12 places her finger over the image of hand 37 on touchscreen 23 and moves her finger on touchscreen 23 in the direction of the arrow to location (X_2 Y_2) - the hand image on LCD 22 follows her finger as described above with reference to Fig. 7. Pipe 35 intersects coordinates (X_2 Y_2) and hence when hand 37 intersects pipe 35 at coordinates (X_2 Y_2) the program being executed in microprocessor 50 in control unit 47 interprets this collision as a command to show hand 37 grasping whatever object is at coordinates (X_2 Y_2) - in this example pipe 35. The polygons which form the image of hand 37 on LCD 22 are then modified by microprocessor 50 (Fig. 4) to show hand 37 grasping pipe 35 on LCD 22. If player 12 implements this action, microprocessor 50 sends data to console 42 where microprocessor 86 (Fig. 16) modifies corresponding polygons which form the image of hand 37 in the generated video images displayed on TV 11 (Fig. 16). Hence, when touchscreen 23 is used to move an object in the

picture on LCD 22 from one LCD location to another location, the resulting action appears on both the LCD 22 and TV screen 56.

The X,Y coordinates in Fig. 11 may be denominated in pixels or millimeters and refer to the visible area of LCD screen 22 and corresponding area of touchscreen 23. Since the picture on LCD 22 is a two-dimensional picture, there is no Z coordinate, although Z may represent a non-spatial variable such as finger pressure. The X,Y coordinates on LCD screen 22 should not be confused with simulated coordinates X,Y,Z in a simulated 3-dimensional world populated with animated characters, a world in which Z represents height.

Fig. 12 illustrates another use of cursor control in a war game where a first human player uses touchpad 24 (Fig. 3) to control cursor 49 on handheld control unit 28 (Fig. 3). He first uses touchpad 24 to position cursor 49 at a map location indicated by the + sign. Then he presses button 14 (Fig. 3) to define the starting point of a line of defense. Then using touchpad 24 to position cursor 49 as shown in Fig. 12, he presses button 14 again to define the end point of the defense line. Control unit 28 then displays a line of dots 30 in Fig. 13 representing a line of soldiers. The first player can also indicate building a barrier across bridge 39 (Fig. 13) using cursor 49 (Fig. 13). Since these tactical moves are displayed only on the first player's control unit, the line of soldiers and the bridge barrier are secret from a second player or players who may falsely assume that the soldiers are deployed elsewhere and bridge 39 is open. If the first player displays the map later, the same line of soldiers 30 and barrier on bridge 39 will continue to appear on the LCD screen of the first player's control unit, but will not be displayed on corresponding maps displayed on control units held by other players.

Fig. 14 illustrates a map with a limited display area 74 that can be scrolled in various directions by using cross-switch

15 to display a different area of the map such as display area 75 which may show greater detail than Fig. 13 on the same size LCD 22. Moving a finger on touchpad 24 or touchscreen 23 may be used in lieu of cross-switch 15 to relocate the display area on a map.

Thus control units with touchpads 24 and LCD screens 22 as illustrated in Fig. 3 are very useful to control a video war game where the battles are displayed on TV screen 56 (Fig. 2) for all players to see, but where tactical moves are planned and executed in secret on handheld control units. Performing the same functions with cross-switch 15 on control unit 44 as in Fig. 2 would typically be less natural, more difficult, and slow.

Fig. 15 illustrates a menu of alternative actions which appears on LCD screen 22 awaiting selection by human player 12. LCD screen 22 is overlaid by touchscreen 23 (Fig. 4) so that the next action for character 18 to perform among these four alternative actions is selected by player 12 touching the touchscreen 23. Character 18 in each of the four action pictures may be the same character, a player controlled character who is controlled by player 12. When player 12 touches one of the four touchscreen areas corresponding to the four pictures in Fig. 15, control unit 28 (Fig. 8) or 47 (Fig. 1) generates data indicating which of the four corresponding locations is selected. Console 42 (Fig. 8) then begins one of the four possible action sequences selectable at the current branch point, i.e. one of the four preprogrammed actions. For control units that have LCD 22 but not touchscreen 23, the procedure described above with reference to Fig. 5 using a cross-switch may be used instead of a touchscreen.

Fig. 15a illustrates a menu of alternative actions which appear on LCD screen 22 as a series of pictures, each picture representing one alternative action for the character to perform. In this example there is no touchscreen 23 overlaying LCD 22 and human player 12 cycles through the series of pictures until the desired action appears on the screen 22 and is then selected.

Fig. 16 is a block diagram of the major components of the exemplary video game system indicated generally at 19 and also shown in Fig. 8 (Fig. 8 and Fig. 16 show different handheld control units). Game console 42 includes a housing indicated by the dashed line in Fig. 16 and shown in isometric view in Fig. 8. Disk 43 is shown outside this housing for clarity, but may be played within the housing. Inside this housing is a small computer consisting of microprocessor 86, RAM 90 for storing video game programs and data, boot ROM 91 for power up and reset and may include an operating system such as DOS, non-volatile EPROM 89, EEPROM, or battery-maintained SRAM for storing digital information that is different for each game console 42, video signal generator 117 (see US patent 6,139,433) for generating composite or separate audio and video suitable for input to TV set 11 or a video monitor (not shown), and peripheral interface chip 88 for sending and receiving digital data to and from handheld control units 44 and 47 (Fig. 1) and control units 28 and 29 (Fig. 8).

For clarity, specialized coprocessors for D/A conversion, audio, or for rendering texture-mapped polygons, terrain rendering, and related graphics processing are not shown.

Disk reader 83 reads digital information from plastic optical disks such as disk 43 in which the digital information is molded and burned. Disk reader 83 reads this digital information from two areas of disk 43: from area 81 and from area 80. In area 81 the digital information is represented as a long spiral track or tracks 82 of microscopic pits that are molded into each disk by a disk manufacturer. Digital information in area 81 includes video game programs and data. Area 80, known as the burst cutting area (BCA), typically consists of a circular series of variable-width bar codes that are burned, melted, or heated by a medium power laser beam into each disk after the disk is molded by the manufacturer. This heating process permanently alters reflectivity of bar-shaped areas of a reflective layer in the

disk. The word "burned" will be used herein to encompass the various methods for placing a substantially unique bar code (for each game product) onto each disk, even though the reflective layer is usually not burned through but merely darkened. More than a hundred patents have been issued for optical disks, BCA, and related technology, such US patent 6,081,785.

In the BCA bar code, each variable width bar represents one bit. The maximum number of bits in the BCA is limited to 1,504 bits (188 bytes) under the current standard. Eighty BCA bits are sufficient for authentication because in the exemplary embodiment, the BCA bits are a block-encrypted cipher of a serial number and another number used for verifying authenticity.

Much of the digital information read from disk 43 by disk reader 83 is controlled by security processor chip 84 so that chip 84 can block processing of video game data from unauthorized disks. An exemplary security chip 84 is further detailed in Fig. 17.

Fig. 17 shows the video game system of Fig. 16, but with more details on security chip 84 and processing of BCA data. Security chip 84 is a microcontroller with an on-chip microprocessor (not shown) for executing instructions from an on-chip ROM (not shown) to perform functions shown in Fig. 17.

If all authenticating data were in the BCA bar code burned into each disk, then software pirates could easily defeat authentication by copying BCA's from authentic disks to non-authentic disks. It is therefore preferable for disk reader 83 to distinguish at least two physically different types of authenticating data which are shown in Fig. 17 as burned bar codes 80 and molded lead-in control data track 148. In this example, disk reader 83 accepts data from track 148 only if it a molded track with the standard optical properties of molded pits, i.e. not burned or a writable CD. There are numerous ways of

making bar codes 80 and molded track 148 physically different. A simple way to make them different is to mold control data 148 into the disk during the same manufacturing step that molded area 81. Mere separation of the burned 80 data from the molded 148 data on different optical tracks or writing some of the data onto a magnetic track would provide little security.

In this example, disk reader 83 distinguishes molded data from burned data in the BCA and this is indicated in Fig. 17 by separate lines through disk reader 83, one line from molded control data track 148, a second line from molded program and data tracks 82, and a third line from burned bar codes 80.

In this example, data from molded control data track 148 includes an encrypted hash value 144 computed from game programs and/or data on tracks 82 during manufacturing (discussed below with reference to Fig. 18). This encrypted hash value 144 is encrypted by the game vendor using a non-symmetrical "public key" cryptographic system as a digital signature. RSA, ECC, or other public-key cryptosystems may be used and are typically controlled by a private and public key of about 1,020 bits and typically produce an encrypted ciphertext of more than 1,020 bits. This ciphertext (encrypted hash value 144) is molded into control track 148. MD5, SHA-1 or similar hashing methods may be used to compute the hash value which may consist of 128-bit, 160-bit, or other size binary numbers before being encrypted. Decryption process 107 uses the same cryptographic method to decrypt value 144 under control of "public key" 95 to produce the original hash value 145. In this example there is no need for public key 95 to be revealed to the public.

Data from burned BCA bar codes 80 includes encrypted control record 94. In this example, encrypted control record 94 consists of at least 88 bits and preferably 128 bits and is encrypted by the game vendor using a symmetric block encryption method such as the Data Encryption Standard (DES), AES, or equivalent, so that

changing any one bit of plaintext affects all bits of ciphertext, without providing clues that would lead to discovery of the bit values of the secret key K2 through chosen plaintext attack or chosen ciphertext attack. Secret key K2 is securely stored in security processor chip 84, preferably in EPROM 98, or EEPROM that is physically protected against chip peeling and scanning electron microscopy. Key K2 is not externally readable from chip 84. DES is described in detail in the Federal Register 40FR12134, March 17, 1975. Simplified variations of DES may be used for block decryption process (99 in Fig. 17) and the corresponding block encryption process (147 in Fig. 18).

Block decryption process 99 decrypts encrypted control record 94 under control of secret key K2 (98) to produce a block of decrypted data including serial number 101 and secret key K1 (reference number 100). One-way hashing process 108 calculates a hash value from key 100 hashed together with all or selected portions of the programs and/or data read from tracks 82 into RAM 96.

Processor instructions 106, stored and executed in security chip 84, compare decrypted hash value 145 to calculated hash value 112. If the two numbers are equal, security chip 84 permits further reading of programs and data from disk tracks 82 into RAM 96 for execution by microprocessor 86. If hash values 112 and 145 are different, then process 26 will block further reading of disk 43, perhaps by endless looping.

Block decryption process 99 uses the same secret key 98 for decryption 99 (Fig. 17) as for encryption 147 (Fig. 18). Typically this key 98 is at least 64 bits and preferably 80 bits. In the preferred embodiment, there is not one master key on chip 84, because if it were compromised, perhaps by an employee or contractor of the game vendor, security chip 84 would become useless. Instead, in the preferred embodiment, each security chip includes a table of keys (not shown) so that secret key 98

can be changed in mid production of any game title by changing to a different key in the table. If the key bits in EPROM 98 are intermingled with unused random bits, anybody who accesses the bits will not know which bits are key bits without also reading the on-chip ROM program that knows which bits are key and which are decoys. If key EPROM 98 is mask programmed, that would reduce security of the keys.

Whenever process 99 decrypts encrypted control record 94, one of the decrypted data fields is serial number 101. Therefore in the preferred embodiment, chip 84 includes a process for comparing serial number 101 against a table (not shown) of known invalid serial numbers, i.e. serial numbers that have been found on illegally copied game disks. If serial number 101 is invalid, then process 26 will block further reading of disk 43.

Security chip 84 is designed to authenticate game disks such as disk 43, but not to protect the programs and data on the disk from reverse engineering. In this embodiment, it is assumed that game programs and data on tracks 82 are not encrypted. However, in the preferred embodiment, at least a portion of the programs/data on tracks 82 should be encrypted to deter pirates from bypassing security chip 84. Improvements may be added to security chip 84 to decrypt encrypted programs and/or data and other methods of improving security. The details of security chip 84 are given here only as examples and numerous other designs may be used.

Fig. 18 shows a disk manufacturer's process for writing data onto disk 43. Programs and data 96 are molded as tracks 82 into disk 43 by disk molding process 149. During the same molding process, encrypted hash value 144 is also molded into disk 43 in lead-in control track 148. Encrypted hash value 144 is previously computed by the game vendor as follows: Key K1 (reference number 100) is generated as a random number. One-way hashing process 108 then calculates a hash value 145 from key 100 hashed together

with all or selected portions of the programs and/or data in RAM 96. MD5, SHA-1 or similar hashing methods may be used to compute hash value 145 which may consist of 128-bit, 160-bit, or other size binary numbers. Any attempt to alter even one bit of the hashed programs and/or data will result in a very different hash value 145.

This hash value 145 is then encrypted under control of private key 166 using the same non-symmetrical "public key" cryptographic process discussed above with reference to Fig. 17. The results of encryption process 167 is encrypted hash value 144 which is then molded into control track 148. RSA, ECC, DH, or other public-key cryptosystems may be used for encryption process 167.

Serial number 101 and key K1 (reference 100) are encrypted together (as a block) by block encryption process 147 under control of secret key 98 (key K2) to produce encrypted control record 94. Encrypted control record 94 is then burned into BCA bar codes 80 in disk 43 by BCA burner 150, using a different serial number 101 for each disk 43. This makes the BCA bar code substantially unique for each of the disks.

Fig. 19 shows a record format of exemplary data records use for communication between processor 50 in control unit 28 and microprocessor 86 in console 42 by way of cable 45 or equivalent. Each record 78 consists of several data fields including a control unit identification number so that console 42 will know which control unit generated record 78, a picture serial number so that console 42 will know which video frame is being referred to, and a size factor number so that console 42 will know the degree of enlargement so it can relate LCD screen locations to simulated objects in the picture. Each record 78 has an operation code which specifies the type of data and what type of

processing is to be performed. Examples of operation codes include:

00 initial power up
01 identify location and size factor of displayed picture
02 move object located at (X_1 Y_1) to location (X_2 Y_2)
03 first person approach to object located at (X_1 Y_1)
04 build object id3 between locations (X_1 Y_1) and (X_2 Y_2)
05 change object located at (X_1 Y_1) with object id3
06 destroy objects between (X_1 Y_1) and (X_2 Y_2)
07 show hand grasping object at (X_1 Y_1)
08 show object at (X_1 Y_1) entering object at (X_2 Y_2)
09 enlarge object located at (X_1 Y_1)
10 change camera angle to center on object at (X_1 Y_1)
11 retreat from object at (X_1 Y_1)
12 selection from action menu
13 cancel or undo previous action serial number nnn

Since the above X,Y coordinates typically refer to physical locations (in pixels or millimeters) on LCD 22 and not always to spatial coordinates X,Y,Z in the simulated world of the animated characters, there is no Z spatial coordinate in the Fig. 19 record format. However, if control unit processor 50 (Fig. 4) can convert physical LCD location coordinates into simulated spatial coordinates and send this data to console 42, then the location data in Fig. 19 would change accordingly. If processor 50 can determine the character action corresponding to a LCD location and send this action data to console 42, the Fig. 19 record would include numbers specifying selected actions.

Fig. 20 is a memory map of various programs and data in RAM 53 in control unit 28 (Fig. 4). Many of the functions performed by these programs are combined in the flowchart in Fig. 21.

Fig. 21 is an exemplary flow chart illustrating a sequence of functions performed by some of the programs temporarily stored in RAM 53 in control unit 28. Fig. 21 begins with program

process 60 which executes out of ROM 76 and converts any initial manual inputs into numbers in memory to be sent to console 42. For example, a player may hold down button 14 as he or she turns on electric power to control unit 28 to activate previously stored game status data. Then in program process 61 (operating out of ROM 76) processor 50 sends a power-up data record (operation code 00) to console 42 which requests that console 42 send initial programs (read from disk 43) to control unit 28 for storage in RAM 53. When those programs are stored, processor 50 continues with program 62 which processes picture data records received from console 42.

Process 63 then generates a picture for display on LCD 22 that is a miniature likeness of the TV frame currently displayed on TV screen 56. Process 64 then displays the miniature likeness picture on LCD 22. The control unit program then enters a program loop which checks (decision boxes 65, 66, 67) for any manual input from a cross-switch, joystick, touchscreen, touchpad, or button switches to determine which kind of location data to send to console 42 (boxes 68, 69, 70). Control unit processor 50 then sends a location data record (or other type of record) to console 42. The interrupt feature of processor 50 may be used to insure that loops shown in Fig. 21 do not interfere with other functions performed by processor 50.

Processor 50 in control unit 28 may generate many of the picture sequences with infrequent guidance from console 42, especially during time intervals when the pictures displayed on LCD 22 are not being displayed on TV screen 56. For example in a war game (referring to Figures 12, 13, and 14), strategic and tactical planning may be controlled by each player on separate handheld control units 44 and 47. Because these private pictures and/or words are not shared with other players by way of TV screen 56, there is no need for frequent sending of data records back and forth between control units and console 42 during these private phases of the interactive game. During this

private phase, each control unit acts independently of console 42, executing programs for planning, deployment of soldiers, movement of supplies, building of bridges, destroying enemy barriers, reconnaissance, displaying reports from spies etc, while the TV screen shows generic scenes and information already known to both sides, such as maps of recent battles, or animated characters controlled by other players.

During game phases where the TV pictures are related to the LCD pictures, there will be much sending and receiving of data records between control units and console 42. During these shared phases, console 42 programs in RAM 90 (Fig. 16) determine what is to be displayed on each control unit 28, 44, etc. and generate picture or program data records which microprocessor 86 sends to one or the other control units. When a control unit receives a data record from console 42, decision box 73 transfers control to process 62 which processes the received picture data record. If data records from console 42 contains program instructions, process 62 in this example will load the downloaded program into RAM 53 for execution in the control unit processor 50.

Figures 22 and 23 illustrate the relationship between video pictures on TV screen 56 and a miniature likeness being displayed on LCD screen 22. In Fig. 22 a large detailed picture is being displayed on TV screen 56. If this detailed picture is greatly reduced in size (perhaps by 90%) for display on a small LCD screen 22 on a handheld control unit 28, many of the details may be lost and the miniature picture may become unintelligible.

Fig. 23a illustrates this loss of detail. One way of avoiding this problem is for processor 50 to generate wider lines and other details as in Fig. 23b from compressed data supplied by console 42. The LCD picture 33 in Fig 23b is a miniature likeness for display on LCD 22 and does not have to be an exact copy of the TV screen picture reduced in size. Another method is illustrated in the Fig. 23c picture which consists of

about 250 short line segments that together form a simplified likeness of the picture on TV screen 56 and omits fine textures displayed on TV screen 56. Further simplified pictures may be used on LCD 22.

Fig. 24 shows an exemplary and simplified block diagram of system 19 showing how data flows between console 42 and a handheld control unit 28. When disk reader 83 reads game programs into RAM 90, the programs in this example are of two kinds, console program(s) 151 with associated data, and controller program(s) 152 with associated data. Focusing on the programs, controller program 152 is transmitted to RAM 53 in handheld control unit 28 and executed in microprocessor 50. Console program 151 is stored in RAM 90 and executed by microprocessor 86 which generates animated picture data 146 representing one or more animated characters performing an action. This data stored in RAM 146 is converted to a video signal as described above with reference to Fig. 16. This video signal is passed to TV 11 by way of cable 41 (Fig. 16) and is displayed as animated pictures on TV screen 56. Microprocessor 86 also generates data records which it sends (arrow 154) to control unit 28. An example of a data record 78 is illustrated and discussed above with reference to Fig. 19. Other record formats may be used by programs 151 and 152.

Execution of console program 151 is controlled by data received (arrow 153) by console 42 from microprocessor 50 in control unit 28. Microprocessor 50 receives (arrow 154) the data records received from console 42 and this data affects execution of program 152 in microprocessor 50 which also receives manually entered input signals from cross-switch 15 (only one of the 4 switches is shown), analog joystick 20, touchscreen 23, and/or other manual controls. These input signals result from a human player's decisions based on animated pictures that are displayed on LCD 22 from animated picture data 146 generated by microprocessor 50 executing program 152 in RAM 53. The input

signals also control execution by microprocessor 50 which sends corresponding data records (arrow 153) to console 42.

Fig. 25 is an exemplary flow chart illustrating a sequence of functions performed by some of the programs temporarily stored in RAM 53 in control unit 28 to replay pictures previously displayed on LCD 22. As with Fig. 21 discussed above, Fig. 25 begins with program process 60 which executes out of ROM 76 and converts any initial manual inputs into numbers in memory to be sent to console 42. Then in program process 61 (executing out of ROM 76) processor 50 sends a power-up data record to console 42 (as discussed above with reference to Fig. 21). If decision box 73 determines that a new picture-data record has been received by control unit 28, processor 50 continues with process 62 which processes picture data records received from console 42. From this data, process 63 then generates a picture for display on LCD 22 that is a miniature likeness of the TV frame currently displayed on TV screen 56. Program 159 provides blinking or highlights, if any are specified in the picture-data record, to accent objects (such as 31 on Fig. 1) in the likeness picture. Program 64 then displays the likeness picture on LCD 22. Processes 65, 66, and 67 (discussed above with reference to Fig. 21) then check for player manual input.

Decision box 156 determines if the player has manually selected a blinking or highlighted object. If such an object was not selected, the object is still selectable and the player may want to return to it later using the replay feature detailed here. Decision box 156 then passes control to process 79 which adds a new record to a replay table 165 of data in RAM 53 from which the full-screen picture containing the blinking or highlighted object can be regenerated on LCD 22. A digital pointer (not shown) points to the last (latest) record in table 165. If the object was selected (and therefore no longer blinking or highlighted), decision box 157 determines if the

picture should still be saved in replay table 165 to preserve continuity of motion during later use of the replay feature. For example, data for regenerating one picture per second may be saved in replay table 165. Processor 50 proceeds to decision box 72 in Fig. 25 which loops back to decision box 73.

If decision box 73 in Fig. 25 determines that no picture-data records are pending, processor 50 proceeds to decision box 160 which checks button-switches and other manual inputs to determine if a player has requested the replay option. If yes, process 163 sets a pointer to the beginning (oldest record) of replay table 165 discussed above, and process 158 generates a miniature likeness from data in replay table 165. If decision box 161 determines that the player selected the fast-forward option to return picture-by-picture to the latest likeness picture, process 164 adds 1 (one) to the table pointer which points to the next data record in replay table 165. If decision box 161 determines that the player has not selected either the replay of fast-forward options, control passes to process 65 discussed above.

Fig. 26 illustrates an exemplary game playing session in which two human game players 10 and 12 use two kinds of portable control units. Player 10 uses control units 184 and 185. Player 12 uses control units 191 and 192. Control units 185 and 192 each have one or more joysticks 20 and/or touchpads controlling player-controlled characters in a simulated three-dimensional world displayed on screen 56. Control units 184 and 191 each have an LCD screen on which are displayed pictures of animated characters and other objects, icons, maps, tables, verbal expressions, and/or other visual images, so that player 10 for example can select and control objects, characters, and actions on LCD screen 22 using control members on unit 185 and/or 184. LCD control units 184 and 191 may be placed on a table 187 or other support within easy reach and viewing whenever control units 185 and 192 are in use. Images on each LCD screen are

hidden from other players by the respective housing of LCD control units 184 and 191.

Control units 184, 185, 191, and 192 are connected to game console 42 by electrical cables 45 and 186 or by equivalent wireless data transmission such as radio waves, infrared, and ultrasound.

Operating LCD control units 184 and 191 is similar to operating control units 44, 47, 28, and/or 29 discussed above, except that control members on control units 185 and 192 such as joystick 20 and touchscreen 24 may control visual information on LCD screen 22. This visual information may include cursors, highlighting, scrolling, three-dimensional control of characters in pictures, and other visual information on LCD screen 22 of control units 184 and 191 by way of game console 42. In this example, console 42 has a game program that provides transfer of joystick data from cable 45 to cable 186 or wireless equivalent, to an LCD control unit 184 and/or 191. When a control unit controls an LCD screen on the same or another control unit instead of a TV screen, the LCD visual information is hidden from other players.

Fig. 27 illustrates how a picture displayed on LCD screen 22 may differ from a picture displayed on TV screen 56, depending on the point of view (perspective) from which a simulated "camera" is pointed within a three dimensional world generated by the video game system. In this example, there are two simulated "cameras" 173 and 188. The picture generated from the perspective of camera 188 appears on TV screen 56 and includes, in this example, a side view of player-controlled character 18 running from an attacking dinosaur. From the perspective of camera 173, i.e. from the subjective point of view of player-controlled character 18, camera 173 pointed at variable camera angle 177 (a direction controlled by a human player) views (generates) object 172 which, in this example, is motorcycle 193.

A player can manually change angle 177 to direct camera 173 at other objects that may provide alternative means of escape. The picture generated from the perspective of camera 173 appears on LCD screen 22 in portable control unit 184. The relationships between camera and display screen are indicated by lines of dots in Fig. 27.

Generating a picture from the perspective of a player-controlled character is referred to as the "subjective mode" in US Patent 6,139,433, column 36, in which all camera modes display pictures on a common TV screen for all players to see.

In multi-player games it may be important for each player to conceal from other players any knowledge of which object 172 a player-controlled character 18 is observing, i.e. what image is generated from the point of view of camera 173, especially as camera angle 177 and other directions are controlled privately by a human player. This concealment is achieved in this example by displaying object 172 in a subjective mode on an LCD screen 22 that is hidden from other players by the housing of portable control unit 184 of which LCD 22 is a component (see Fig. 26) or is hidden by the housing of handheld control unit 44 (see Fig. 1).

If object 172 is another player-controlled character, such as an animated character, or a remote-controlled motorcycle 193, or a remote-controlled robot (described below with reference to Figures 30 and 31) the player can relocate the point of view from camera 173 to another camera at the point of view of a newly selected or newly activated player-controlled character represented in Fig. 27 as object 172. The player relocates the point of view by pressing a combination of buttons or other manipulatable device on handheld control unit 185 or LCD control unit 184 or by pointing to object 172 with a cursor or highlight using a manipulatable device or a combination thereof in a control unit or units. When the player relocates the point of

view from character 18 to object 172, the picture displayed on LCD 22 in control unit 184 will be from the point of view of object 172, which in this example may be another player-controlled character. A player can relocate the point of view multiple times through a chain of player-controlled characters and objects using this method.

In Fig. 27, when a player selects a point of view 173 and a direction angle of view 177 for display on LCD 22, the player may zoom-in on object 172 by manipulating a control member on a handheld control unit, so that the image of object 172 generated for display on LCD 22 is enlarged in a manner similar to enlarged image 25 discussed above with reference to Fig. 1. This enables a player to examine object 172 in greater detail on LCD 22 or on TV screen 56. The player may also zoom-out by manipulating the same control member which causes the picture on LCD 22 to cover a broader viewing angle 222 and field of view in Fig. 27.

Fig. 28 illustrates a menu of alternative virtual "buttons" displayed on LCD screen 22 for controlling points of view, joystick selection, and other alternatives in games where each player may control multiple characters, robots, or similar objects. Robot characters are explained below with reference to Fig. 29. Each alternative can be manually selected by using cross-switch 15 (Fig. 5), or joystick 20 or 21, touchpad 24, touchscreen 23 (Fig. 3), or other manipulatable control member on a control unit to move a displayed cursor, or highlight, or other indicator to a "button" for selection. In Fig. 28, button 190 for "point of view" has been selected by the player and is highlighted. A new character, such as a robot, can be activated by selecting the "activate character" button in menu row 189.

Fig. 28a illustrates a picture menu displayed on LCD screen 22 whenever the "point of view" button 190 is selected in Fig. 28. Pictured in this menu are five characters as examples controlled by the player viewing this LCD screen 22, except for

the dinosaur which is a non-player character whose point of view can be used by any player. The player viewing the picture menu in Fig. 28a selects the character whose point of view is to be displayed on LCD screen 22. In this example, the player has selected the player-controlled character "flying robot 1" (indicated by shading) by using the cross-switch or other control member on control unit 184 or 185 (Fig. 26), so that pictures displayed on LCD 22 will be generated from flying robot 1's point of view.

Each player has one and possibly more than one player-controlled character whose points of view may be used for viewing on the player's LCD control unit. Since a player may be controlling more player characters/robots than joysticks in this example, a joystick that controls camera angle is assigned to the character selected in the Fig. 28a menu.

Activation of a character or characters would normally cause automatic assignment of a joystick to control movements of that character(s) and a second joystick to control point of view. If a player prefers a different joystick or touchpad or other control member, selecting the "select joystick" button in menu 189 in Fig. 28 displays a different list (not shown) on LCD 22 for assigning a joystick or other control member to an active character or group of characters which the player wants to actively control. Assignment of joysticks is also discussed below with reference to Fig. 32 and the "Robot Control Panel".

Fig. 29 illustrates a map view of a video game in which two player-controlled characters (animated character 17 and robot character 155) are controlled by the same human player, although in some embodiments not all functions of both characters can be controlled simultaneously. If more than one player is playing this game, each player can control multiple characters individually and in groups. In the Fig. 29 example, animated player-controlled character 17 is standing at the entrance to a

cave tunnel 176 shown in cross-section with walls 170. From the point of view of character 17, object 172 is displayed on LCD 22 (not shown in Fig. 29) when her "camera" 173 is pointed at angle 177. When her camera 173 is pointed toward the entrance to cave 176, character 17 cannot see deep into the cave and her body is too large to crawl into the cave. To explore the cave, a human player may activate a small character such as land-crawling robot 155 illustrated in Fig. 31 and indicated by the circled R in Fig. 29, with point of view selected as discussed above with reference to Fig. 28a. The player controls movements, directions, and point-of-view perspectives of robot camera 175 like any other player-controlled character using a control unit, such as LCD control unit 44 in Fig. 1, or joystick control unit 185 in Fig. 26, or touchpad 24 in control unit 28 in Fig. 3.

In the Fig. 29 example, character 17 is displayed on TV screen 56 and may appear motionless or as manipulating a robot-control unit at the entrance to cave 176 while robot 155 explores the cave. The point of view of robot 155 is from camera 175 which is controlled by the same player as camera 173. Camera 175 is pointed at object 171 which may be hidden treasure in this example that is accessible only by a small robot. The player then has a problem of removing the treasure from cave 176 using grippers 181 on robot 155 illustrated in Fig. 31.

In Fig. 29, when a player selects a point of view 175 and a direction of view for display on LCD 22, the player may zoom-in on object 171 by manipulating a control member on a handheld control unit, so that the image of object 171 generated for display on LCD 22 is enlarged in a manner similar to enlarged image 25 discussed above with reference to Fig. 1. This enables a player to examine object 171 in greater detail on LCD 22 or on TV screen 56. The player may also zoom-out by manipulating the same control member which causes the picture on LCD 22 to cover a broader field of view in Fig. 29.

Figures 30 and 31 illustrate simulated "radio-controlled" robots 174 and 155 that a player can activate and send on a mission in this exemplary game. A flying robot is illustrated in Fig. 30 and has a camera 179 which transmits pictures to LCD 22 or TV 56, at the player's option. Camera 179 can also be pointed vertically downward for aerial reconnaissance to produce maps such as map area 75 illustrated in Fig. 14. Flying robot 174 also has a speaker 183 for whispering secret messages to another character or to the player, or for making public (in the game world) announcements. Robot 174 also has one or more claws or grippers 181 for picking up objects, for putting objects together or for taking them apart, for carrying objects, and for deactivating objects (defusing a bomb, for example).

Fig. 31 illustrates a land crawling robot 155 (used in Fig. 29) with caterpillar tread 180, camera 179, headlight 182 (useful in dark cave 176), and one or more claws or grippers 181. Other robot types (not shown) may be submersible robots, spaceship repair robots, microscopic robots, robots with specialized tools, materials, and skills (for example a welding robot), a humanoid robot with legs, arms, eyes, etc, or other robot types. Other non-robot player-controlled characters may be used (that is animated characters), such as a chemistry character that can generate food from minerals, or a financial character that can find money), and other character types.

Fig. 32 illustrates a control panel displayed on LCD 22 on control unit 184 whenever a player selects the "robot control panel" button shown in Fig. 28. This Robot Control Panel links cross-switches, joysticks, touchpads, and other control members on a specific handheld control unit to movements of robot arms, grippers 181, headlight 182, caterpillar tread 180, and other tools attached to each robot, some of which are shown in Fig. 31. Each control unit has a number indicated in the number box which is constant if a player has only one control unit, but may be

variable if a player has two or more control units, as illustrated in Fig. 44. In the Fig. 32 example, the control panel allocates a small number of joysticks (two in this example, a left joystick and a right joystick) to the many possible movements of the robot arms and grippers (fourteen different movements in this example). When a player allocates one joystick to each pair of movements, manually controls that movement, and then allocates that joystick to a different pair of movements, a player can control all of the necessary movements through a series of allocations of one or two joysticks. Each joystick, touchpad, etc. can be allocated to more than one pair of movements and one of the button switches on a control unit can cycle through the allocations. For example, if joystick 21 in Fig. 3 is allocated to right arm extension of robot 155 in Fig. 31, pressing button 14 may cause joystick 21 to be reallocated to point-of-view control. Pressing button 14 again may cause joystick 21 to be reallocated to left arm extension, and so on, as previously set up on the Robot Control Panel in Fig. 32. Pictorial representations such as icons, arrows, pictures of robot components, and others pictures may be used on LCD 22 on control unit 184 instead of the word descriptions used in the Fig. 32 example.

A control panel may also be used to allocate joysticks to control movements of humanoid characters which have more possible movements (shoulder, torso, and others). An example of humanoid characters manipulating an iron pipe into position for use as a prybar is illustrated in Fig. 2.

Fig. 33 illustrates a picture menu displayed on LCD screen 22 on control unit 184 whenever a player selects the "activate character" button shown in Fig. 28. A player-controlled character may be activated by pointing to its picture on LCD 22 with cursor 59, or by highlighting it and selecting it, or by using a touchscreen. A character is then activated, and its face or body will be shown in picture menus or lists as an active

character. This picture menu may also be used to deactivate a character by moving cursor 59 to the "D button" for Deactivating. The "A button" is for Activating an inactive character in this example. If more than one copy of a character is activated, for example a general purpose character such as a robot, a number on each robot icon distinguishes the various copies.

A group of characters may be activated by indicating the number of characters in the number box in Fig. 33 by positioning cursor 59 on the box and pressing one of the buttons n times on a control unit. Other methods may be used to indicate the number of characters in a group. In this example, an active group such as a platoon of soldiers responds to joystick control as if the group were a single player-controlled character. Each group or group leader also has a point of view that can be used to display objects on LCD 22 as seen from the point of view of the group. In Fig. 27, for example, player-controlled character 18 may be a group of characters which sees object 172 from camera 173 at angle 177 from the point of view of the group in this example.

Fig. 34 illustrates a picture menu of assignable tasks displayed on LCD screen 22 on control unit 184 whenever a player selects the "assign task button" in menu 189 in Fig. 28. The player first indicates the desired task on the Fig. 34 task menu using cursor 59 or other indicator, and then indicates which active character-controlled character is to perform a task by selecting a character on the Fig. 33 character menu.

Unlike the picture menu illustrated in Fig. 15 which shows alternative actions for a player-controlled character that remains a player-controlled character after an action is selected, each task illustrated in Fig. 34 is a preprogrammed activity that temporarily releaves a player of the burden of micromanaging every movement of his player-controlled character. If a player selects a Fig. 15 action or enters a new level in this example, the player-controlled character is still player

controlled and the player uses an analog joystick to control every movement of the player-controlled character. Only one player-controlled character can be simultaneously controlled by each player in this example.

But when a character(s) is assigned a task, the character(s) automatically perform the preprogrammed assigned task so that the player-controlled character is temporarily like a non-player character in a task-controlled mode until the task is completed or interrupted. That makes it practical for a player to supervise several player-controlled characters that need not be acting as a group, by giving each character limited and temporary autonomy.

For example, robot 155 illustrated in Fig 31, is represented by the circled R in Fig. 29 and is a player-controlled character whose detailed movements while searching for treasure through cave 176 may be controlled by a player using an analog joystick 20 on control unit 185 (Fig. 26). The player may also have an option of assigning a cave-exploring task to robot 155. Robot 155 then becomes a task-controlled character preprogrammed to search cave 176 without getting lost (or maybe getting lost and radioing for help). Robot 155 reports back to the player whenever the robot finds something that may be treasure. Since the robot need not be programmed to be sufficiently "smart" to distinguish a pot of gold from an old rubber tire, the human player may be an active decision maker in the treasure search even though the player is not controlling every robot movement.

The point of view of robot 155, during the cave-exploring task, is represented in Fig. 29 as camera 175 which is the point of view from which object 171 is generated for display on LCD screen 22 on the player's control unit. This point of view feature may be active even when a player-controlled character temporarily becomes task-controlled.

Movements of task-controlled characters are preprogrammed but may adjust to the environment in an intelligent manner, avoiding obstacles, picking up or changing the correct objects in the correct manner, returning with the correct objects, etc. Task-controlled characters may be displayed on TV screen 56, but this feature can be overidden by the player for secret tasks by selecting a "secret" mode (not shown) so that assignment and performance of tasks and other functions are displayed on LCD screen 22 rather than TV 56, and so that other player(s) will not know what task is being performed by each character.

The number of characters assigned to a task may be automatically determined by the game program. A group of characters may be a team and each team member may perform a different sub-task in coordination with other team members. The player may then supervise the team as a whole and not control each team member.

Each task has a beginning time when the task is initiated, an ending time when the task is terminated, and while active between those points in time, a task does something useful to or with an object or objects, each of which may be another character or an inanimate object, and may be the task-controlled character itself. For example a task may require a character to move an object to a different location in the simulated world, and that object may be itself. A task may require a character to find a missing object, to construct an object, to disassemble an object, to repair an object, to gather objects together, to cook food, to play music, etc. LCD 22 may display a picture menu (not shown) of objects that are appropriate for each task in the context of the game at the time a task is assigned.

Whenever an assigned character encounters a problem, preprogrammed or otherwise, the character reports its status on the LCD screen 22 of the player who activated it, indicated perhaps by the character's face blinking, and control reverts to

the player to move the character as a player-controlled character and to control its tools using joysticks or other manipulatable devices on a control unit, or to reassign a joystick if they are currently assigned to other characters. A player may select a task in a picture menu using a cross-switch, but if a player uses a controller with a touchscreen 23 (Fig. 4), then tasks may be selected with a finger touch just as actions are selected in Fig. 15. A mouse-like touchscreen 23 or touchpad 24 (Fig. 3) would make assigning tasks easier than using cross-switch 15.

Whenever a task is assigned to a newly activated character or team of characters, the game program may automatically provide the tools and supplies appropriate for the task, unless searching for such tools and supplies is another preprogrammed task that a player controls. For example, if the task is changing a car tire as in Fig. 34, a land crawling robot 155 with a tire-changing toolkit would be assigned by the game program, rather than assigning a flying robot. Delivering a message over water or mountains would be a task assigned to flying robot 174. A player would specify which message should be delivered and specify the character to whom the message should be delivered, but the direction, speed, and altitude of flight of the flying robot from second to second would be generated by the game program unless the player prefers to do detailed movement control. Other player-controlled characters may be assigned tasks which are performed concurrently or sequentially. After a generic character completes it's task or mission, the generic character may be inactivated automatically.

Fig. 34a illustrates a screen for display on LCD 22 that lists all available inactive tasks with a short description (pehaps a few sentences) that can be used to assign a task or mission that is difficult to define with a picture as in Fig. 34. Additional screens on LCD 22 may be used to assign a complex task to a single player-controlled character or to a group of characters by using word descriptions. The "active tasks" button

in the Fig. 34a example causes display on LCD 22 of a screen showing the faces of each character and a picture or description of the task or tasks assigned to each character.

The "interrupt" button temporarily stops an active task and returns to the player-controlled mode so that a player can move the character out of trouble, make a decision, and label an object as dangerous so the character will stay away from it. For example in the cave exploration, the robot character may encounter a set bear trap not knowing what it is. The player can interrupt the task before the object traps the character, and then continue with the task by selecting the "resume" button which returns the character to the task-controlled mode. If a player changes his/her mind about activating a task, he/she can purge it from the queue of active tasks by selecting the "cancel" button.

There may also be preprogrammed interrupts when a character encounters a situation where motion control or a decision by a human player is needed. When that happens, the task is still active, but is automatically put on hold temporarily and processing continues in the player-controlled mode.

Fig. 35 is an exemplary flow chart illustrating a sequence of program decisions and functions or processes performed by some of the programs temporarily stored in RAM 90 (Fig. 24) in the game console and/or RAM 53 in a portable control unit and executed in the respective microprocessors 86 and/or 50. Fig. 35 begins a double loop, the outer loop iterating through each of the human players, and an inner loop iterating through each of the active tasks, for each player.

In each of these loops, decision box 201 checks if the task-controlled mode is on, and if not, decision box 200 checks if an "assign task" has just activated a new task as illustrated in Figures 34 and 34a. If not, then the system is still in the

player-controlled mode and program process 213 performs the normal player-controlled processes. If "assign task" (illustrated in Figures 34 and 34a) has just activated a new task assigned to a character, process 209 initiates the task-controlled mode for the new task. If the task-controlled mode is on as determined by decision boxes 201 or 200, decision box 202 then checks if the current task is completed. If yes, program process 210 terminates the completed task, shows a "task-completed" status message on the LCD control unit of the player currently being processed, changes to player-controlled mode, and continues in this mode with process 213.

If the current task is not completed, decision box 203 checks if the current player has selected the "resume" button illustrated in Fig. 34a. If yes, program process 211 resumes the task that had earlier been put on hold by program process 212 and the normal task-controlled process 214 continues. If "resume" was not selected, decision box 204 checks if the current task is on hold. If yes, the system is temporarily in player-controlled mode and processing continues with process 213. If the current task is not on hold, decision box 205 checks if the current player has selected the "interrupt" button illustrated in Fig. 34a. If yes, program process puts the task on hold and processing continues in the player-controlled mode with program process 213. If the "interrupt" button was not selected, the normal task-controlled process 214 continues.

After either process 213 or 214 has been done for this one iteration, decision box 208 checks for end of game. If yes, the Fig. 35 process exits. If not end of game, decision box 207 checks if there are any remaining characters in the inner loop. If yes, the inner loop continues with the next character at decision box 201. If all characters have been processed in this inner loop, the inner loop ends and decision box 206 checks if there are any remaining players in the outer loop. If yes, the outer loop continues with the next player at decision box 201.

If all players have been processed in this outer loop, the outer loop ends and the Fig. 35 process exits.

If a player selects the "cancel" button illustrated in Fig. 34a, the current task is interrupted as described above with reference to decision box 205, but instead of putting the task on hold as in process 212, the task is removed from the current character's list of tasks and the normal player-controlled process 213 continues.

Fig. 36 illustrates an exemplary game playing session in which two human game players 10 and 12 use two kinds of portable control units. Player 10 uses control units 184 and 185. Player 12 uses control units 191 and 192. Control units 185 and 192 each have one or more joysticks 20 and/or touchpads controlling player-controlled characters in a simulated three-dimensional world. Control units 184 and 191 each have an LCD screen on which are displayed pictures, verbal expressions, and/or other visual images so that each player can select and control objects, characters, and actions on the LCD screen. LCD control units 184 and 191 are mounted on opposite sides of a support assembly 194 with a vertical shield that holds the LCD controllers in a vertical position and blocks each player from seeing the other player's LCD screen. Support assembly 194 may be placed on a table 187 or other support within easy reach and viewing whenever control units 185 and 192 are in use. Support assembly 194 in Figures 36, 37, and 38 prevent LCD control units 184 and 191 from falling over as they may do when sitting on table 187 with wires attached as shown in Fig. 26.

Fig. 37 is a detailed view of support assembly 194 which securely holds two portable LCD control units 184 and 191 for use in a two player game. Support assembly 194 consists of a vertical member 197 mounted at the base with a dual hinge assembly 198 hinged with two horizontal coplaner base boards 199 which can rotate into vertical positions parallel with vertical

member 197 for compact storage and packaging. Alternatively, instead of hinge assembly 198, a rectangular socket (not shown) can be attached to one or two horizontal base boards 199 for receiving vertical member 197 in the socket, as illustrated in US Patent 4,022,473.

Each base board 199 has an attached latching leaf spring 196 for securing the bottoms of portable LCD control units 184 and 191. On each side of vertical member 197 are mounted two hooks 195 that engage two holes in the top of each control unit. Hooks and holes for control unit 184 are not shown. When a player attaches a portable control unit 191 to the assembly illustrated in Fig. 34, the player first places control unit 191 under the two hooks 195 which enter the two holes in the control unit. The player then rotates control unit 191 toward vertical member 197 until the bottom of control unit 191 presses against latching leaf spring 196 which bends slightly downward until control unit is in position for spring 196 to click back into its original position, thereby securing control unit 191 against accidental rotation. Leaf spring 196 may be movable horizontally to adjust the viewing angle of LCD screen 22 on each control unit.

The player then plugs electrical cable connector 40 into a data socket in control unit 191. Connector 40 is wired to cable 186 which enters a hole in hollow vertical member 197 and exits a second hole near hinge assembly 198. Cable 186 connects to console 42 as shown in Fig. 26. Cable 186 also connects to control unit 184 through a similar hole (not shown) in vertical member 197.

Vertical member 197 and base boards 199 are shown larger than required for support of control units so that maps, tokens, and other board game components may be placed on base boards 199 and vertical member 197 and hidden by vertical member 197 from view by the other player in a two-player game.

Fig. 38 illustrates an alternative design of support assembly 194 which securely holds two portable LCD control units 184 and 191 for use in a two player game. Support assembly 194 consists of a vertical member 197 mounted at the base to a single horizontal base board which may include a socket (not shown) for receiving vertical member 197.

Fig. 39 illustrates a map view of a video game in which player-controlled character 18 is displayed on TV screen 56 as a running man, viewed (indicated by the short line of dots) from the point of view of "camera" 188. From the point of view of character 18, object 172 is initially viewed from "camera" 173 as described above with reference to Fig. 27. In Fig. 39, a human player can view object 172 from different points of view represented by cameras 175 and 215 and can view object 172 at angles 177 and 216 respectively. The angle 177 or 216 at which object 172 is viewed is variable and is manually controlled by the player. Camera 175/215 may also zoom in or zoom out on object 172 so that object 172 appears larger or smaller on LCD 193. Object 172, which in this example is motorcycle 193, may be displayed on LCD 22 (indicated by the long line of dots) to prevent other players from seeing object 172 on TV screen 56. Object 172 may also be displayed on TV screen 56.

A human player controls movements, directions, zoom, and point-of-view perspectives of cameras 175 or 215 using a directional input control member, such as cross-switch 15 on LCD control unit 184, or joystick 20 on control unit 185 in Fig. 26, or touchpad 24 or touchscreen 23 in control unit 28 in Fig. 3 or similar control members. Object 172 may be viewed from any angle such as 177 or 216 horizontally and in three dimensions from above and from below (not shown), where the viewing angle is centered on or near object 172 or any other object selected by the player. The point of view of camera 175/215 may move around object 172 so that LCD 22 displays object 172 from many different points of view and directions in the simulated three-dimensional world.

Fig. 40 is a memory map of programs for performing functions described above with reference to Fig. 39, and data processed by those programs.

Fig. 41 illustrates a video game in which two or more players can trade tools and other objects they have acquired during the game; for example, a key for opening a door or treasure chest in the simulated world, scissors for cutting a rope to a required length, reward items such as coins, and other objects. A picture menu or word menu of objects controlled by each player is displayed on their respective LCD control units 44 and 47, or on TV 11. Each player selects one or more objects they are offering to trade and these objects may be displayed on TV screen or video monitor 56 or on other player's LCD control units. Negotiating a trade may be conducted verbally, but when two players reach an agreement, a program in video game console 42 acts as an escrow agent, thereby insuring that each player gives up control of the object they agreed to trade and receives control of the object they expect in return.

As illustrated in the middle of Fig. 41, LCD control units 44 and 47 of players participating in a trade display the objects being traded so that a third or fourth player may not know what objects were traded. If two players approve of trading the displayed objects by manipulating control members on their respective hand-held control units, a program in the console system closes the trade by updating game records to reflect the change of ownership and displays on LCD control units 44 and 47 a picture of the object received by each trader, as illustrated at the bottom of Fig. 41.

Fig. 42 is a three dimensional graph illustrating cartesian coordinates (X_1 Y_1 Z_1) of an exemplary camera 175 and coordinates (X_2 Y_2 Z_2) of an exemplary object 171 being photographed by the simulated camera. See examples in Fig. 29 and 39. Polar coordinates would also be an appropriate equivalent. For

clarity, coordinates are not shown for camera 215 which may be the same as camera 175 but at different location in the generated three dimensional world.

Fig. 43 illustrates an exemplary game playing session in which human game player 10 manipulates control members on control unit 185 while viewing pictures, maps, and other visual images displayed on two or more LCD screens 22 on portable game control units 184 and 191. These control units are connected by cable, wireless, or other data transmission means to game console 42. Player 10 controls character 17 in a simulated three-dimensional world generated by console 42 and transmitted via cable 41 to video display unit 11 for display on screen 56. Player 10 further manipulates control members on control unit 185 or 184 or 191 to select alternative views of the the simulated world for display on LCD 22 in units 184 or 191 or both. Control unit 185 or 184 or 191 transmits control data to console 42 which responds by transmitting data to control unit 184 or 191 or both that specifies an image for display on the respective LCD screen 22.

By having two or more LCD display devices 22 each showing different locations in the simulated world that are different than the view displayed on video screen 56 and viewed from different angles, the player can select and monitor trouble areas in the simulated world similar to a security guard monitoring closed-circuit television pictures from security cameras. A program in console 42 may cycle through several views selected by player 10 for display in succession on one or more LCD screen 22. A map of one part of the simulated world may be displayed on one LCD control unit, while a picture of a portion of the simulated world is displayed on another LCD 22, and while a different map or a different picture appears on video screen 56. Control units 184 and 191 are supported by table or shelf 187.

Fig. 44 illustrates an exemplary game playing session in which at least two human game players 10 and 12 both control the same player-controlled character, in this example land crawling robot 155. Player 10 manipulates control members on control unit 185 while viewing closeup pictures of his portion of robot 155 displayed on LCD screen 22 on portable game control unit 184. Likewise, player 12 manipulates control members on control unit 192 while viewing closeup pictures of her portion of robot 155 displayed on LCD screen 22 on portable game control unit 191. Each player controls different functions of robot 155.

For example, player 10 may control robot arm movement and movement of caterpillar tread 180 (see Fig. 31), while player 12 may control several gripper 181 movements. Players may specify which controller and which joystick is to be used to control each robot function by inputting settings on the Robot Control Panel described above with reference to Fig. 32.

Player-controlled characters controlled by more than one player may also include animated humanoid, animated animal, animated alien, and other types of characters. One player may control movement of a character on land and inside buildings, while another player may control movement of the same character in tunnels and while the character is flying or swimming, as examples. One player may control a character walking and running and point-of-view selection, while another player may control the same character jumping and fighting and weapon selection, as examples.

Fig. 45 illustrates an exemplary adapter 218 (drawn with thick lines) for use with portable game unit 219 (drawn with thin and dashed lines). Adapter 218 provides additional control members that may be unavailable on portable game unit 219 such as joysticks 20 and 21, button switch 14, adjacent button switches, and touchpad 24. Game unit 219 slides into adapter 218 where it is secured by data communication cable 221 or additional spring

latch (not shown in Fig 45). The totality of functions provided by game unit 219 inside adapter 218 is similar to functions provides by control unit 28 described above with reference to Fig. 3, with the possible exception of touchscreen 23 and speaker 27 which are not shown in Fig. 45. Fig. 45 is divided by orthographic projection into front view in Fig. 45a, top view in Fig. 45b, and right side view in Fig. 45c.

Fig. 46 illustrates electronic circuitry inside adapter 218 described above with reference to Fig. 45. Data communication cable 221 plugs into portable game unit 219 and enters adapter 218 where it connects directly or indirectly to data communication cable 45 which transmits data to video game console 42. Microprocessor 50, which in this example includes on-chip ROM and RAM, collects manually entered control data from switches 14, from analog joysticks 20 and 21, and optionally from touchpad 24. Peripheral interface chip 88 converts this control data from microprocessor 50 to serial data which may be multiplexed with serial data from portable game unit 219. The functions of peripheral interface chip 88 and microprocessor 50 may be combined in one chip. Serial data from video game console 42 and cable 45 pass to portable game unit 219 by way of cable 221. Optionally, data from video game console 42 and cable 45 may pass to peripheral interface 88 to enable functions of adapter 218, for example, to enable an LED (not shown) in adapter 218 to indicate that data connections are operational.

As used herein, the terms "video screen" and "display unit" include the display area of a television screen, computer monitor, video monitor, RGB monitor, CRT, flat screen, and the like. The term "video" includes composite, non-composite, RGB, monochrome, color, analog, digital, and MPEG video, and the like. The term "molded" includes injection molded, pressed, stamped, and other disk manufacturing methods. The term "three-dimensional world" includes two-dimensional worlds. The word "camera" as used herein denotes a point of view from which a real camera would see the generated picture at a specified angle. The term "microprocessor" may include two or more cooperating processors.

The term "likeness" includes pictures that have a similar character performing a similar action, even though there are noticeable differences in resolution, texture, terrain, and other details. The term "program" as used herein may consist of more than one loadable module and includes executable instructions and any data and addresses that are typically part of a program module or modules. For simplicity, animated characters and other objects may be represented herein as circles and other shaped figures which do not imply that they would be so represented on a TV screen or LCD. Characters, objects, and situations in a video game are not limited to those depicted herein for illustrative purposes.

The term "LCD" (liquid crystal display) has been used herein as an illustrative example of any discrete display apparatus having discrete picture elements. Other discrete display technologies may be substituted for LCD technology. The expression "portable control unit" is used herein to mean a control unit that may be held in a person's hand, but is not necessarily being so held during operation of a video game.

Although I have described my invention with a degree of particularity in connection with what is presently considered to be the most practical and preferred embodiment, it is to be understood that the present disclosure has been made only by way of example and that my invention is not to be limited to the disclosed embodiment, but on the contrary, is intended to cover various modifications, arrangements, steps, and components included within the spirit and scope of the appended claims.

Reference Numbers in Drawings

10 human player
11 television (TV) set or video monitor
12 human player
13 LCD screen
14 push button
15 cross-switch (directional pad)
16 memory cartridge
17 player-controlled character
18 player-controlled character
19 video game system generally
20 joystick assembly
21 joystick assembly
22 LCD screen
23 touch screen
24 touch pad
25 small area of LCD screen
26 process of stop reading disk
27 speaker in control unit
28 handheld control unit with handles
29 handheld control unit with handles
30 representation of military barrier
31 clue object
32 picture of emotional face
33 picture on LCD screen
34 picture on LCD screen
35 picture of iron pipe
36 picture of player character's hand
37 picture of player character's hand
38 infrared communication unit
39 representation of a bridge
40 electrical connector
41 cable linking game console to TV
42 video game system console
43 optical disk

44 handheld control unit
45 cable linking control unit to console
46 handheld control unit
47 handheld control unit
48 highlighted image
49 cursor
50 microprocessor in control unit
51 touchpad and/or touchscreen processor
52 memory security processor
53 random access memory (RAM) in control unit
54 game product number
55 process of checking authenticity of disk
56 TV screen
57 push-button
58 dinosaur's foot
59 cursor
60 program process
61 transmission of data
62 program process
63 program process
64 displaying an LCD picture
65 program decision
66 program decision
67 program decision
68 program process
69 program process
70 program process
71 transmission of data
72 program decision
73 program decision
74 map display area
75 map display area
76 read only memory (ROM)
77 memory map of programs and data
78 location data record
79 program process

80 burst cutting area (BCA) of disk
81 program and data area of disk
82 tracks molded into disk
83 optical disk reader
84 security processor
85 speaker in TV set
86 microprocessor in console
87 electrical connector
88 peripheral interface processor
89 EPROM or EEPROM
90 RAM in console
91 boot ROM
92 address bus
93 data bus
94 encrypted control record
95 "public" key
96 unencrypted programs and/or data in RAM
97
98 secret key K2
99 process of block decryption
100 secret key K1
101 disk serial number
102 process of validating disk serial number
103
104
105
106 process of authenticating programs/data
107 process of RSA decryption
108 process of calculating hash values
109
110
111
112 hash value
113
114
115

116
117 video signal generator
118 video game system generally
119 LCD driver circuit
120
140
141
142
143
144 RSA encrypted hash value
145 hash value
146 animated picture data
147 process of block encryption
148 lead-in control information
149 process of molding disk
150 process of burning BCA into disk
151 console program
152 controller program
153 data transmission
154 data transmission
155 land-crawling robot
156 program decision
157 program decision
158 program process
159 program process
160 program decision
161 program decision
162 program decision
163 program process
164 program process
165 table of data in RAM
166 RSA private key
167 RSA encryption process
168
169
170 cave wall

171 generic object
172 generic object
173 point of view "camera"
174 flying robot
175 point of view "camera"
176 cave, tunnel, or maze
177 "camera" angle
178 radio antenna
179 camera
180 caterpillar tread
181 gripper or claw
182 head light
183 speaker
184 control unit with LCD
185 control unit with joystick
186 cable linking portable game and console
187 table or shelf
188 "camera"
189 menu items
190 menu items
191 control unit with LCD
192 control unit with joystick
193 picture of motorcycle
194 support assembly in general
195 mounting hooks
196 latching leaf spring
197 vertical assembly
198 hinge assembly
199 base board
200 program decision
201 program decision
202 program decision
203 program decision
204 program decision
205 program decision
206 program decision

207 program decision
208 program decision
209 program process
210 program process
211 program process
212 program process
213 program process
214 program process
215 "camera"
216 "camera" angle
217 memory map of programs and data
218 adapter
219 portable game unit
220 shoulder button switch
221 cable linking adapter to game unit
222 field of view angle